MITRE CWE and CERT Secure Coding Standards

Robert C. Seacord, Software Engineering Institute [vita[1]]

Robert Martin, MITRE Corporation [vita[2]]

2010-06-15 L3 / D/P[3]

This article describes the relationship between the Common Weakness Enumeration (CWE) and CERT Secure Coding Standards.

Common Weakness Enumeration

The Common Weakness Enumeration (CWE) is a unified, measurable set of software weaknesses that enables the effective discussion, description, selection, and use of software security tools and services that can find these weaknesses in source code and operational systems. The CWE also enables better understanding and management of software weaknesses related to architecture and design. It enumerates design and architectural weaknesses as well as low-level coding and design errors.

CERT Secure Coding Standards

CERT[9], part of Carnegie Mellon University's Software Engineering Institute, is developing secure coding standards for commonly used programming languages such as C, C++, and Java through a broad-based community effort that includes members of the software development and software security communities. CERT secure coding standards include guidelines for avoiding coding and implementation errors as well as low-level design errors.

Well-documented and enforceable coding standards are essential to secure software development. Coding standards encourage programmers to follow a uniform set of rules and guidelines determined by the requirements of the project and organization, not by the programmer’s familiarity or preference. Once established, these standards can be used as a metric to evaluate source code (using manual or automated processes) for compliance with the standard.

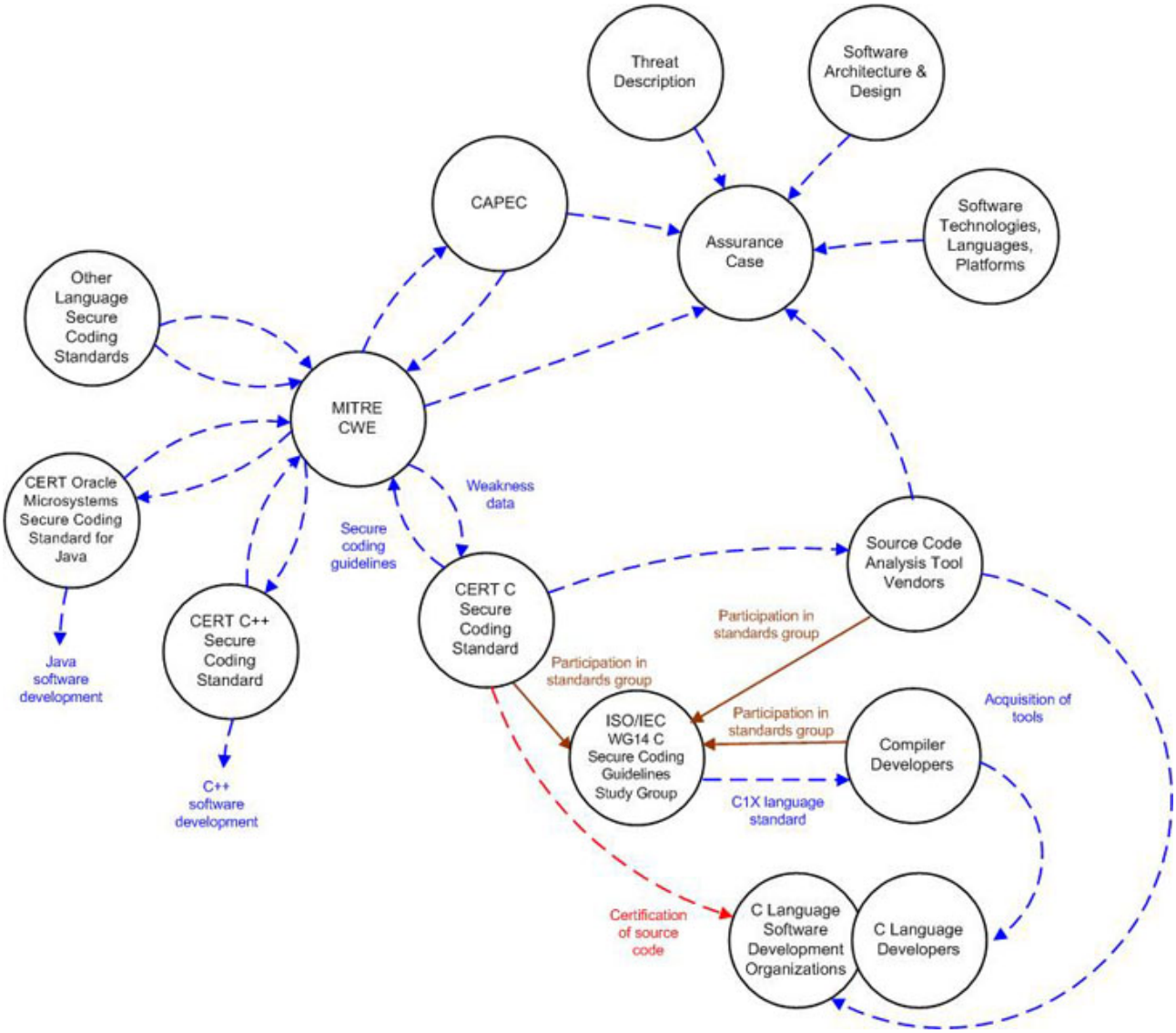
CERT-CWE Relationship

The CWE and the CERT secure coding standards perform separate but mutually supportive roles. Simply stated, the CWE provides a comprehensive repository of known weaknesses, while CERT secure coding standards identify insecure coding constructs that, if present in code, could expose a weakness or vulnerability in the software. Not all weaknesses enumerated in the CWE are present in any particular secure coding standard because not all weaknesses are present in each language and because the CWE also includes high-level design issues. Not all CERT secure coding guidelines are mapped directly to weaknesses in the CWE because some coding errors can manifest in ways that do not directly correlate to any given weakness. Both tools are necessary in evaluating the security and safety of software systems.

1. http://buildsecurityin.us-cert.gov/bsi/about_us/authors/274-BSI.html (Seacord, Robert C.)
2. https://buildsecurityin.us-cert.gov/adm-bsi/about_us/authors/1159-BSI.html
4. #dsy1158-BSI_CWE
5. #dsy1158-BSI_CERT secure coding
6. #dsy1158-BSI_CERT-CWE relationship
7. #dsy1158-BSI_CERT-CWE mappings
8. #dsy1158-BSI_references
9. CERT is registered in the U.S Patent and Trademark Office by Carnegie Mellon University

Figure 1[10] illustrates the software development aspect of the software assurance ecosystem supported by the CWE and the CERT secure coding standards. Related views could be constructed to show how these tools support acquisition, education, and research.

Figure 1. The CERT-CWE landscape



CERT-CWE Mappings

The CWE contains multiple views, the most important of which are the full dictionary view, the development view, and the research view. The CWE-734 view enumerates weaknesses addressed by the *CERT C Secure Coding Standard* [Seacord 2009[12]] and includes 103 out of the 799 total CWEs. Developers can fully or partially prevent the weaknesses identified in this view if they adhere to that standard. Because not all CERT guidelines map to specific weaknesses, the CWE-734 view is incomplete.

In addition, developers can use a CWE coverage graph to determine which weaknesses are not directly addressed by the standard. Making that determination can help identify and resolve remaining gaps in training, tool acquisition, and other approaches for reducing software weaknesses.

10. #dsy1158-BSI_figure1
12. #dsy1158-BSI_seacord09

Guidelines in the *CERT C Secure Coding Standard* are cross-referenced with CWE entries. These cross-references are created only for guidelines that, if violated, directly contribute to the referenced weakness. Similar mappings will be created for other CERT coding standards once the coding standards are completed.

References

[Seacord 2009] Seacord, Robert C. *CERT C Secure Coding Standard.* Addison-Wesley Professional, 2009. www.securecoding.cert.org/confluence/display/seccode/CERT+C+Secure+Coding+Standard[14]

Carnegie Mellon Copyright

1. mailto:permission@sei.cmu.edu